REMARKS

In the Office Action of April 7, 2004, claims 11-20 directed to a system, method and software for providing persistence of complex data objects and their data relationships were pending, and claims 1-10 were cancelled. Unfortunately, because of an initial failure of the Examiner to understand the significance of the meaning of the term "CDOG" or complex data object graph and "ejb" or enterprise java bean, as defined in the specification, the Examiner has continued to objected to the terms CDOG and ejb in the claims, and has now cited a totally new reference that does not anticipate the CDOG term while inappropriately rejecting all of the pending claims as being anticipated, as follows:

1. Claims 4-6 were rejected due to informalities because of the abbreviated terms "ebj" and "CDOG" in the claims.

2. Claims 11-20 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent 6,633,889 issued to Dessloch et al. ("Dessloch").

With the above amendment, claims 11-20 (all of the prior pending claims) have been canceled without prejudice, and new Claims 21-30 have been added that correspond to the amended subject matter of cancelled claims 11-21. The word order of the claims has been changed without changing the substance to further clarify the claims (as suggested by the Examiner during an informal interview

during July 2005) and the abbreviations "ebj" and "CDOG" have been expanded. Such amendments are made to expedite the prosecution of this case, only. Applicant reserves the right to assert the deleted or canceled subject matter in this case, or in a continuing application. Reconsideration of this subject matter is respectfully requested in view of the traversal of the prior art rejection included herein. After entry of the above amendment, Claims 21-30 are active in the case. The above rejections are addressed in part by the present amendments and are otherwise traversed by the arguments that follow.

THE AMENDMENTS

New base claim 21 has been amended to clarify that "ejb" and "CDOG" are abbreviations for "enterprise java bean" and "complex data object graph" and to arrange the word order to better understand and distinguish the formal requirements of a computer system that are required by the patent office from the improvement in the art that is claimed. In short, each of claims has been amended to provide claims that are a computer system loaded for software that has the capability to accomplish the desired result as a result of an interface and the software that utilizes either the local or distributed computer system.

Moreover, the software claims relate to software that is defined as having executable features that accomplish tangible results when loaded on a computer system. As recited in the claims, the computer program may be loaded on a local or distributed computer system and executed to utilize that system to create or maintain transparent persistence of complex data objects and their relationships as a complex data object graph. The meaning of transparent persistence is defined in the present specification in that the data from complex data objects and their relationships within an application are being persisted without the application being in communication with the software that is persisting all or a portion of the complex data objects and their relationships as a complex data object graph.

The Examiner is to understand that a complex data object graph (as defined in the specification and currently claimed) is defined as a term which includes both data objects and the definitions of their relationships to other data objects (i.e., their "links" between each other). This is to be clearly and easily distinguished from a collage of the data from multiple data objects into a single data object or virtual single data object. In such an architecture that is dramatically different from the current system, data from multiple objects are incorporated into a single software

component or "super data object" whose data can be accessed by a software program running within the system.

Unlike the present architecture that persists both data objects and their relationship definitions (i.e., persistence as a complex data object graph), when a "single software component", "super data object" or a "virtual data object" is utilized for persistence the data persisted from within such a super data object is persisted independent of its relationships and without any knowledge of the relationships being defined or required. Moreover, the virtual data object persistence is "static" rather than "dynamic" since the virtual data object must be updated or entirely re-written by a software programmer, if the relationships between objects in a complex data object graph (schema) are changed.

Another weakness of such a "collage of data objects" single component "virtual data object" system is that it requires the loading of data from one or more relational data sources, conversion of the software into data objects without defining any links between the objects, and then software for lifting of certain pieces of data from individual data objects to incorporate that data into a single software component, or virtual data object. Such multiple steps can be slow, they

are prone to error can be very complex to maintain. However, when data is persisted as a complex data object graph and links between data objects are changed between the data objects in that schema (complex data object graph) there is no single software virtual data object to create. The stateful generic session bean have the registered name of the complex data object graph (CDOG) will simply access the CDOG model definition for any changes in the definitions in data objects and/or their relationships (links) and persist both the update/changed/deleted data objects and their relationships definitions (links) to other objects.

THE 102(A) REJECTION UNDER DESSLOCH

Claims 1-3 and 7 were rejected under 35 U.S.C. § 102 (e) as being anticipated by U.S. Patent 6,633,889 issued to Dessloch et al. ("Dessloch"). Unfortunately, although Dessloch NEVER mentions or defines the important term "CDOG" (complex data object graph) as defined in the specification and clarified in the prior response, the Examiner persists in wrongly rejecting the present claims as being anticipated as if Dessloch had included such a term. Accordingly, the final rejection is entirely inappropriate and should be withdrawn. Details reasons are point out below:

As pointed out by the abstract of Dessloch, as well as in the main claims of Dessloch (see, Claim 1(c) of Dessloch), regarding the Dessloch technology...the main idea of Dessloch is provide persistence of "pieces" of data from multiple data objects by forming a collage of data portions from multiple data objects into a "single virtual data object" that consolidates the data from those multiple data objects that are registered in the Dessloch system. As noted above, such a collage of data objects into a single component known as a "virtual data object" requires multiple processing steps and multiple data layers removed from the data source, all of which can be accessed by a software component within the system and used as a virtual "object data source" buffer. This design may be a data caching shortcut that requires less memory than keeping all of the data objects in the memory, but the overhead and chance for error may very well negate such a design. In any event, a collage of pieces of data into a single large virtual data object does not anticipate a complex data object graph (CDOG) persistence design.

In the present case the CDOG is registered with the system...not just data objects being registered. The system is dynamic, since if any of the data objects or the definitions of their relationships with other objects (links) are changed the "hot"

update occurs on the fly without any need to update a composite "virtual data object" as required by Dessloch. As described in the present specification and as claimed, the generic session bean (software) with which the CDOG is registered within the claimed computer system can efficiently can track any changes during the object access session and can be set to lazy load into the memory or into a cache only the data objects and their relationships that have relevant data, which needs to be accessed for a given transaction.

A further distinction is the generic session bean that tracks both data objects and their relationships in order to persist the CDOG, is ejb that is modified for a specific unit of work. Any CDOG can be registered and its objects and relationships definitions can be persisted without the need to update a specially adapted ejb for accessing only an exhaustively defined single virtual data object collage of just data as described by Dessloch. Such is very different from the computer or distributed computer network system presently claimed that can provide transparent persistence of complex data objects as a CDOG.

An object model of complex data objects and their relationships to other objects within the Java object programming sense relates to Java Classes, for example, that may wrapper data and the logical relationships between different Java Classes in their data, such as 1-1, 1-M (1-Many) and M-M. Logical relationships between complex data objects (CDO) in a Java Programming Application may be created or be change "on the fly" (dynamically) completely independently with respect to the hardwired nature of a specifically adapted "virtual single data object" that was created for a specific application. Such parameters and dynamic changes to the CDOG in the Java Software Application that may be running need to be persisted to a data store, particularly if someone else wants to use updated data or relationships. Therefore, the generic ejb stateful session bean (as shown in Example 4 and loaded in a computer system as described in the present specification) monitors changes to data objects and complex data objects of one or more applications that is running on the system and dynamically and transparently persists such changes as a CDOG or a portion of the CDOG without the need for an application to be aware of the persistence or the need for the application to request the system to provide it with such persistence as a CDOG.

Therefore, Dessloch does not even begin to teach or suggest the present invention, much less anticipate it. Accordingly, this ground of rejection is overcome and should be withdrawn.

Moreover, since there is no ancillary reference cited that would teach the equivalence of the Dessloch "single virtual data object" system to the presently claimed computer system that can persist data objects and their relationships as a CDOG, the present claims are also not obvious over the Dessloch disclosure.

CONCLUSION

Accordingly, applicant respectfully submits that the above objections and rejections have been overcome, and should be withdrawn. In view of the amendments (including corrections of informalities) and remarks, the present application is believed to be in condition for allowance. Based upon the aforementioned comments and amendments, it is urged that the claims are in condition for allowance, as is the remainder of the subject patent application. Favorable reconsideration is respectfully requested.

Should the Examiner have any questions, comments, or suggestions, or should issues remain, he is respectfully requested to contact applicant's

representative with the associated power of attorney (see attached) to the undersigned by telephone at 925 594 0900 for a prompt and satisfactory resolution.

In the event that this amendment is not deemed to place this application in condition for allowance, applicant hereby requests a conditional request for continued examination and the required fee may be charged to the deposit account of file for this application.

Respectfully submitted,
Lev Intellectual Property Consulting

Robert G. Lev
Registration No. 30,280

By
J. G. Mullins
Registration No. 33,073

Date: August 8, 2005

4766 Michigan Boulevard
Youngstown, Ohio 44505
Telephone: (330) 759-1423
Facsimile: (330) 759-4865